AMENDMENTS TO THE CLAIMS:

Please cancel without prejudice claims 1-4, 6-8, 12, 14-30 and amend claims 5, 9, 13, 31, 32 and 36 as follows:

This listing of claims will replace all prior versions, and listings, of claims in the application:

1. (cancelled).

2. (cancelled).

3. (cancelled).

4. (cancelled).

5. (currently amended) A method ~~according to claim 4~~of processing data comprising:

processing a function using a processor operable to perform a plurality of functions, said processor having interrupts enabled; receiving an interrupt at said processor during processing of said function at a point at which a portion of said function has been processed;

suspending processing of said function;

accessing at least one control parameter, said at least one control parameter indicating whether processing of said function should be resumed from a point where it was interrupted or whether said function should be repeated from a start of said function following said interrupt such that said portion of said function that has already been processed is processed again;

following completion of said interrupt continuing processing of said function either at said start of said function or at said point at which it was interrupted dependent upon said control parameter; and

controlling said processor to store a restart address at which said processor should continue processing in dependence upon said at least one control parameter, wherein said step of controlling said processor to store a restart address at which said processor should continue processing is dependent upon both said at least one control parameter and said further control parameter, said restart address being said start of said function if said control parameter indicates that said function is to be repeated and said further control parameter indicates said function to have idempotence or an address of said fix-up routine to be performed before said function is restarted if said further control parameter indicates said function not to have idempotence, wherein if said at least one control parameter indicates that said function is to be repeated following said interrupt said method comprises an additional step of:

accessing a further control parameter, said further control parameter being indicative of whether said function has idempotence or not; and

following completion of said interrupt continuing processing of said function at a start of said function if said further control parameter indicates said function to have idempotence, or a fix-up routine to be performed before said function is restarted if said control parameter indicates said function not to have idempotence.

6. (cancelled).

7. (cancelled).

8. (cancelled).

9. (currently amended) A method ~~according to claim 4~~of processing data comprising:

processing a function using a processor operable to perform a plurality of functions, said processor having interrupts enabled; receiving an interrupt at said processor during processing of said function at a point at which a portion of said function has been processed;

suspending processing of said function;

accessing at least one control parameter, said at least one control parameter indicating whether processing of said function should be resumed from a point where it was interrupted or whether said function should be repeated from a start of said function following said interrupt such that said portion of said function that has already been processed is processed again;

following completion of said interrupt continuing processing of said function either at said start of said function or at said point at which it was interrupted dependent upon said control parameter; and

controlling said processor to store a restart address at which said processor should continue processing in dependence upon said at least one control parameter, wherein if said at least one control parameter indicates that said function is to be repeated following said interrupt said method comprises an additional step of:

accessing a further control parameter, said further control parameter being indicative of whether said function has idempotence or not; and

following completion of said interrupt continuing processing of said function at a start of said function if said further control parameter indicates said function to have idempotence, or a

fix-up routine to be performed before said function is restarted if said control parameter indicates said function not to have idempotence, wherein said further control parameter indicates said fix-up routine should be processed prior to restarting said function, said fix-up routine being operable to restore a state of said processor such that said function can be restarted and have idempotence and to disable interrupts during its processing.

10. (original) A method according to claim 9, wherein on receipt of said interrupt said processor processes said fix-up routine prior to handling said interrupt.

11. (original) A method according to claim 9, wherein following completion of said interrupt, said processor goes to an address at which the processor switched modes, and following switching modes, said processor is operable to process said fix-up routine logic prior to restarting said function.

12. (cancelled).

13. (currently amended) A method ~~according to claim 7~~of processing data comprising:

processing a function using a processor operable to perform a plurality of functions, said processor having interrupts enabled; receiving an interrupt at said processor during processing of said function at a point at which a portion of said function has been processed;

suspending processing of said function;

accessing at least one control parameter, said at least one control parameter indicating whether processing of said function should be resumed from a point where it was interrupted or

whether said function should be repeated from a start of said function following said interrupt such that said portion of said function that has already been processed is processed again;

following completion of said interrupt continuing processing of said function either at said start of said function or at said point at which it was interrupted dependent upon said control parameter; and

controlling said processor to store a restart address at which said processor should continue processing in dependence upon said at least one control parameter;

controlling said processor to retrieve stored data relating to a restart address at which said processor should continue processing in dependence upon at least one control parameter, following completion of said interrupt, continuing processing of said function from the stored restart address, wherein said processor is operable in a plurality of modes, said method comprising the additional steps of:

prior to initiation of processing of said function, switching said processor to a mode in which interrupts are automatically disabled on entry to said mode;

storing an address at which said processor switched mode; and

on initiating said function, said function controlling said processor to enable interrupts, wherein said mode is a monitor mode, said processor being operable in a plurality of domains comprising a secure domain and a non-secure domain, such that when said processor is executing a program in a secure domain said program has access to secure data which is not accessible when said processor is operating in a non-secure domain, switching between the domains only being possible when said processor is operating in monitor mode.

14. (cancelled).

15 (cancelled).

16. (cancelled).

17. (cancelled).

18. (cancelled).

19. (cancelled).

20. (cancelled).

21. (cancelled).

22. (cancelled).

23. (cancelled).

24. (cancelled).

25. (cancelled).

26. (cancelled).

27. (cancelled).

28. (cancelled).

29. (cancelled).

30. (cancelled).

31. (currently amended) A computer program product comprising a computer readable storage medium containing computer readable instructions that when executed are operable to control a computer including a processor including ~~comprising~~:

function logic ~~operable to control~~ for controlling a processor to perform a function: and

disable interrupt logic ~~operable to control~~ for controlling said processor to disable interrupts; wherein a first portion of said function logic operable to control said processor to perform a first portion of said function, which has idempotence such that it does not alter a state of any part of the processor which would affect the repeatability of the function and is operable to be executed by said processor before said disable interrupt logic, and a final portion of said function logic operable to control said processor to complete said function is operable to be executed after said disable interrupt logic is executed.

32. (currently amended) A computer program product according to claim 31, further comprising control parameter logic ~~operable to control~~for controlling said processor to write a control parameter, said control parameter indicating that following an interrupt said function should be restarted, said control parameter logic being operable to be executed before said first portion of the function logic.

33. (original) A computer program product according to claim 32, further comprising:

a further portion of said function logic and further control parameter logic; wherein

said further portion of said function logic is operable to be executed after said first portion of said function logic and before said interrupt disable logic, said further portion of said function logic not having idempotence; and

said further control parameter logic is operable to control said processor to write a further control parameter, said further control parameter indicating that following an interrupt a fix-up routine should be run prior to restarting said function.

34. (previously presented) A computer program product according to claim 33, further comprising fix-up routine logic, said fix-up routine being operable to be processed following indication by said further control parameter and, on processing of said fix-up routine, restores a state of said processor such that said function logic can be restarted and have idempotence and wherein said fix-up routine is operable to disable interrupts during said processing.

35. (original) A computer program product according to claim 31, said computer program product further comprising interrupt enable logic, said interrupt enable logic being operable to be performed before said function logic.

36. (currently amended) A computer program product ~~according to claim 31~~ comprising a computer readable storage medium containing computer readable instructions that when executed are operable to control a computer including a processor to include:

function logic for controlling said processor to perform a function; and

disable interrupt logic for controlling said processor to disable interrupts, wherein a first portion of said function logic operable to control said processor to perform a first portion of said function, which has idempotence such that it does not alter a state of any part of the processor which would affect the repeatability of the function and is operable to be executed by said processor before said disable interrupt logic, and a final portion of said function logic operable to control said processor to complete said function is operable to be executed after said disable interrupt logic is executed, wherein said first portion of said function logic comprises logic to control said processor to store data to a stack but not to update a stack pointer, and said final portion of said function logic operable after said disable interrupt logic comprises logic operable to update a stack pointer with respect to data stored to said stack during execution of said first portion of the function logic.